

Development of a Vehicle Model/Simulation Evaluation Tool

J. Gavin Howe
Jeffrey P. Chrstos
Systems Technology, Inc.

Richard Romano
RealTime Technologies, Inc.

James O’Kins
U.S. Army TACOM

Copyright © 2007 SAE International

ABSTRACT

As part of the evaluation of vehicle simulation models, a vehicle dynamics engineer typically desires to compare simulation results to test data from actual vehicles and/or results from known, or higher fidelity simulations. Depending on the type of model, several types of tests and/or maneuvers may need to be compared. For military vehicles, there is the additional requirement to run specific types of maneuvers for vehicle model evaluations to ensure that the vehicle complies with procurement requirements. A thorough evaluation will run two different categories of tests/maneuvers. The first category consists of laboratory type tests that include weight distribution, kinematics and compliance, steering ratio, and other static measures. The second category consists of dynamic maneuvers that include handling, drive train, braking, ride, and obstacle types. In this paper, a process for proper evaluation of vehicle simulation models is presented. A method for evaluating simulation results from different simulation programs is also presented.

INTRODUCTION

Some of the earliest ground breaking work in vehicle dynamics was developed by Olley [Ref. 1] in the mid 1940’s. This work describes such important vehicle characteristics as cornering force, aligning force, camber torque, camber thrust, traction and braking, wheel hop, shimmy, weight distribution, under-steer, Ackerman steer, transient conditions, and more. Vehicle simulation followed in the 1950’s with the best example probably being the seminal work by Segal [Ref. 2]. Vehicle dynamics and simulation continues to be important today.

In 1990, Heydinger, et. al. [Ref. 3] presented a methodology for validating vehicle dynamics simulation

that compared vehicle simulation results to physical testing. They offered the following definition for simulation validation:

“A... ..mathematical model... ..will be considered to be valid if, within some specified operating range of a system, a simulation’s predictions of a system’s responses of interest to specified input(s) agree with the actual physical system’s responses to the same input(s) to within some specified level of accuracy”

In 1994, Bernard and Clover wrote a paper that discussed a validation process for computer simulation of vehicle dynamics [Ref. 4]. In this paper they suggest that three separate questions need to be addressed:

- 1) Is the model appropriate for the vehicle and maneuver of interest?
- 2) Is the simulation based on equations that faithfully replicate the model?
- 3) Are the input parameters reasonable?

The U.S. Army uses physical testing and simulation of vehicles in their procurement process [Ref. 5]. They use a wide range of physical test and simulation to evaluate proposed vehicles. The automotive and commercial vehicle industry also relies heavily on physical testing and simulation in the evaluation of the products they develop.

This paper presents the development a vehicle model/simulation tool called the Model Post Processor (MPP). This tool allows a vehicle dynamicist to select vehicle models from a variety of simulation programs and to evaluate/compare/contrast them using static vehicle metrics and/or using the results from dynamic vehicle maneuvers to evaluate their simulation and/or

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 08 NOV 2007		2. REPORT TYPE N/A		3. DATES COVERED -	
4. TITLE AND SUBTITLE Development of a Vehicle Model/simulation Evaluation Tool				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) J. Gavin Howe; Jeffrey P. Chrstos; Richard Romano; James O’Kins				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000				8. PERFORMING ORGANIZATION REPORT NUMBER 18434	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR’S ACRONYM(S) TACOM/TARDEC	
				11. SPONSOR/MONITOR’S REPORT NUMBER(S) 18434	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited					
13. SUPPLEMENTARY NOTES Presented at SAE 2008 World Congress, April 14-17, 2008, Detroit, MI USA, The original document contains color images.					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT SAR	18. NUMBER OF PAGES 11	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

model. A vehicle model is a parameter set used to describe a vehicle for a given simulation program.

This tool allows the user to compare different vehicle models for the same simulation, vehicle models for different simulations, and to compare vehicle models to physical test data. Having the ability to compare models for use with a single simulation type allows a vehicle dynamicist to make comparisons of different vehicles or comparisons of the same vehicle with minor modifications. Having the ability to select model(s) for a given simulation and comparable test data for a vehicle that the model is based on allows the vehicle dynamicist to determine how well a model simulates the actual vehicle. Having the ability to select models of the same vehicle, but for different simulation types allows a vehicle dynamicist to see how well a lower fidelity model compares to a higher fidelity model. This can be particularly important when running simulations in real time (like when running on a driving simulator) where a lower fidelity model might be based on a higher order model.

The MPP has two subcomponents: Dynamic Vehicle Metrics (DVM) and Consistency Metrics (CM). The DVM consists of a wide range of dynamic vehicle maneuvers and the CM consists of a set of quasi-static vehicle tests. The details of the DVM and CM are discussed later in this paper.

The basic flow of information from the different components is shown in Figure 1. The MPP allows the user to select Simulation and Models for evaluation. This information is then sent to the DVM or CM where Tests and Maneuvers are selected. The basic control inputs for the selected Tests and Maneuvers along with the Simulation/Model selections are then sent to an appropriate Simulation Wrapper. The Simulation Wrapper then creates refined control inputs for the particular Simulation being run. The basic simulation output is then sent back to the Simulation Wrapper that creates refined simulation output that the DVM or CM then uses to create a Report from the analyzed results.

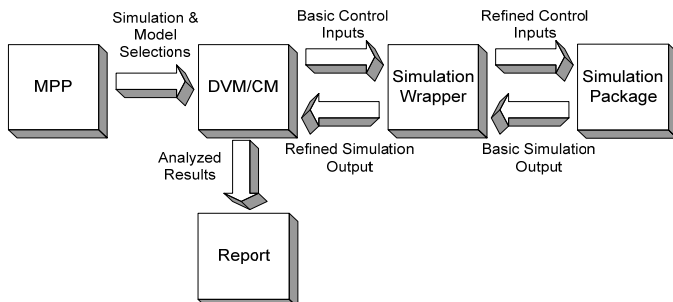


Figure 1 - MPP Information Flow Diagram

It is important to note that there is a separate Simulation Wrapper for each Simulation package. This makes adding additional simulation packages to the MPP relatively simple. The main MPP/DVM/CM needs very

little modification to call the Simulation Wrappers for any new Simulation Package that a user may wish to add.

The development of this tool required the development of directory structures to properly organize different types of data or information, naming conventions for file and data structures, basic control inputs or command files, and simulation wrappers for various simulations. A set of Tests for the CM and Maneuvers for the DVM also had to be selected. The user selectable inputs to define the parameters for each Test/Maneuver and the types of analysis to be performed for each Test/Maneuver had to be determined.

MPP, CM, AND DVM GUI

The MPP, CM, and DVM GUI are shown in the Appendix (Figure 2, Figure 3, and Figure 4).

MPP

The MPP GUI (Figure 2) allows the user to choose up to eight simulation/model selections. Each simulation/model selection has three pop-up menus for "Simulation", "Model", and "Additional Parameters" selection. Simulation lists the currently available simulation programs, Model lists the vehicle models available for a selected Simulation, and Additional Parameters lists the potential additional parameter files for the selected Model.

One of the Simulation options is Test Data. If the Test Data option is made, then the user can select results from actual physical testing to compare against simulation results.

Once the simulation program(s) and vehicle model(s) have been selected, the user can then choose to evaluate the selected combinations using the CM or DVM by selecting the appropriate button which will display the associated GUI.

CM

The CM GUI (Figure 3) allows the user to select quasi-static tests to evaluate the Simulation/Model selections made on the MPP GUI.

Once the user has selected a test, the defining metrics are displayed and the user can enter values for these metrics. Once the defining metrics are entered, the user can then add the test to a list of tests that can then later be run.

Among other settings, the user can select the report type, file names, output units, and fit data option. The report type options are XML, Word, or Matlab Figures. The user can also select a report title that is used to manage the file names for the generated test runs and report. Metric or English output units can be chosen by the user. Regardless of output unit type, the simulation

results are saved in a metric format, but if English units are selected, the units in the generated report (figures and tables) will be displayed in English units. The user can choose to fit the data with linear/cubic fitting routines based on the type of data being analyzed.

Once the user has selected the tests to be run and the other available settings, they can run them by selecting the Run CM push button.

DVM

The DVM GUI (Figure 4) allows the user to select dynamic maneuvers to evaluate the Simulation/Model selections made on the MPP GUI. The features/operations on the DVM are very similar to those for the CM.

TESTS AND MANEUVERS

The CM is used to run quasi-static tests on a vehicle model while the DVM is used to run dynamic maneuvers. A list of selected Tests for the CM and a list of Maneuvers for the DVM are given below as well as some details for running some of the particular Tests/Maneuvers.

TESTS AVAILABLE FROM THE CM

The CM is primarily designed to simulate a Kinematics and Compliance (K & C) test rig. It also has tests designed to get the steering ratio and the basic weight distribution of the vehicle.

A total of eight tests are currently available on the CM: Static Test, Steering Ratio Test, Kinematic Heave, Kinematic Roll, Longitudinal Compliance, Lateral In-Phase Compliance, Lateral Out-of-Phase Compliance, and Aligning Moment Compliance.

The Static Test simply drives the model in a straight line at a very slow speed so the basic weight distribution for the vehicle can be determined. From this test the track widths, wheelbase, ground reactions, total mass, sprung mass, and distances from the center of gravity (cg) to each axle can be determined.

For the Steering Ratio Test, the steering wheel is driven from +/- a steering magnitude input by the user and the wheel forces are set to 0 to simulate the vehicles being on frictionless plates.

The K & C tests are run as a simulation of a physical K & C test rig, where the vehicle chassis is constrained in all 6 degrees of freedom by virtual spring/dampers. For kinematic tests, vertical motions are applied to the tire ground contacts to exercise the suspension while the horizontal tire forces and moments are controlled to be zero. For compliance testing, lateral and longitudinal forces and aligning moments are applied to tire contact patches while the virtual ground plane is held fixed.

The virtual restraint system is three linear spring/dampers and three rotary spring/dampers acting at the vehicle sprung center of gravity. The stiffness of the linear springs are set to allow 0.0254 mm (0.001 in) deflection, Δ_{\max} , under a load equal to the total vehicle weight. The rotary springs are set to allow 0.0254 mm (0.001 in) deflection, Δ_{\max} , when a load equal to the total vehicle weight is applied to a single wheel. The equations used are below:

$$\begin{aligned} K_x = K_y = K_z &= \frac{W \cdot \text{MaxLoad}}{\Delta_{\max}} \left(\frac{N}{m} \right) \\ Km_x &= \frac{W \cdot \text{MaxLoad} \cdot 0.5 \cdot TW}{\tan^{-1} \left(\frac{\Delta_{\max}}{0.5 \cdot TW} \right)} \left(\frac{N \cdot m}{rad} \right) \\ Km_y = Km_z &= \frac{W \cdot \text{MaxLoad} \cdot A}{\tan^{-1} \left(\frac{\Delta_{\max}}{A} \right)} \left(\frac{N \cdot m}{rad} \right) \end{aligned} \quad (1)$$

Where:

Variable	Description	Units
K_x	Longitudinal restraint stiffness	N/m
K_y	Lateral restraint stiffness	N/m
K_z	Vertical restraint stiffness	N/m
W	Total vehicle weight	N
MaxLoad	Multiplier of total weight to set maximum applied load	-
Δ_{\max}	Allowable deflection at max loading	m
Km_x	Rotary stiffness about longitudinal axis	Nm/rad
Km_y	Rotary stiffness about lateral axis	Nm/rad
Km_z	Rotary stiffness about vertical axis	Nm/rad
TW	Front axle track width	m
A	Distance from sprung c.g. to front axle	m

Acting in parallel with each spring is a linear viscous damper. The damper is added to provide numerical stability with the damping ratio, ζ , set to 0.3. The damping values are set to be:

$$\begin{aligned}
B_x = B_y = B_z &= 2 \cdot \zeta \sqrt{M_T \cdot K_x} \left(\frac{N \cdot s}{m} \right) \\
Bm_x &= 2 \cdot \zeta \sqrt{I_{XS} \cdot Km_x} \left(\frac{N \cdot m \cdot s}{rad} \right) \\
Bm_y &= 2 \cdot \zeta \sqrt{I_{YS} \cdot Km_y} \left(\frac{N \cdot m \cdot s}{rad} \right) \\
Bm_z &= 2 \cdot \zeta \sqrt{I_{ZS} \cdot Km_z} \left(\frac{N \cdot m \cdot s}{rad} \right)
\end{aligned} \tag{2}$$

Where:

Variable	Description	Units
B_x	Longitudinal restraint damping	N-s/m
B_y	Lateral restraint damping	N-s/m
B_z	Vertical restraint damping	N-s/m
M_T	Total vehicle mass	kg
ζ	Damping ratio	-
Bm_x	Rotary damping about longitudinal axis	N-m-s/rad
Bm_y	Rotary damping about lateral axis	N-m-s/rad
Bm_z	Rotary damping about vertical axis	N-m-s/rad
I_{XS}	Sprung mass roll inertia	Kg-m ²
I_{YS}	Sprung mass pitch inertia	Kg-m ²
I_{ZS}	Sprung mass yaw inertia	Kg-m ²

MANEUVERS AVAILABLE FROM THE DVM

The DVM can run a wide range of dynamic maneuvers that include handling, drive train, braking, ride, and obstacle types. The list of Maneuvers currently available from the DVM are Slowly Increasing Steer, J-Turn, Swept Sine, Straight Line Acceleration, Straight Line Deceleration, Fishhook, Slowly Increasing Brake, Straight Line Braking, Trapezoidal Bump, Pothole, Half Round, Washboard, RMS Course, and Drawbar Pull.

The types of output for each maneuver are varied and extensive. As an example, the Slowly Increasing Steer maneuver has time series plots for steering wheel angle, throttle, speed, lateral and longitudinal acceleration, yaw rate, roll rate, roll angle, side slip, and load transfer. Plots of understeer gradient, load transfer, and roll gain as a function of lateral acceleration and plots of lateral acceleration and yaw rate gain as a function of steering wheel angle are also available.

Each maneuver has a different set of plots/analysis. Some more unique plots/analysis available are yaw rate, lateral acceleration, and roll rate transfer functions for the Swept Sine maneuver and absorbed power calculations for the RMS courses.

There is also Test Data Maneuver on the DVM. If Test Data is selected, a Test Data Selection panel (directly

below the Maneuver Selection and Maneuver List panels) is displayed. The user can select a Browse push button to open a file selection box that can be used to find an appropriate Test Data file that can be used to drive the simulation. In the Test Data Options panel, which replaces the Maneuver Inputs panel, the user can select the type of plots they wish to make (Plot Selection) and the method for driving the simulations (Drive Selection). The Drive Selection options allow the user to either drive the simulation with Speed control or Throttle control. If Speed control is chosen, the simulation is driven with the Test Data speed channel and if the Throttle Control is chosen, the simulation is driven with the Test Data throttle position channel.

IMPLEMENTATION DETAILS

To implement the MPP, several details needed to be addressed including directory structures, file naming conventions, output data structures, basic command files to be sent to simulation wrappers, and reporting options.

DIRECTORY STRUCTURES

The base directory that the MPP operates from has the following structure:

MPP

- CMSimulationResults
- DVMSimulationResults
- Models

The MPP, DVM, and CM functions need to be able to initiate and make calls to each simulation program that is available in the pull-down menus given on the MPP GUI. These functions also have to be able to load and run the associated vehicle models.

The directory tree for the storage of the associated vehicle model files comes under the major heading "Models" which has headings for each of the simulation program type with sub-headings for each model. The following directory tree for three simulation programs each having two vehicle models is given as an example:

Models

- SimCreator
 - HMMWV
 - STRYKER
- DADS
 - HMMWV
 - STRYKER
- VDANL
 - Taurus
 - WB67

The files required to run each vehicle model need to be stored in the appropriate directory sub-heading.

The simulation program output data are stored under the major heading “DVM (or CM) SimulationResults” with headings and sub-headings for each simulation program and vehicle model, and sub-sub-headings for the report title name selected on the DVM or CM GUI. If the report title is “TestRuns”, and the SimCreator HMMWV and STRYKER models are run, then the directory tree will have the following appearance:

DVMSimulationResults

- SimCreator
 - HMMWV
 - TestRuns
 - STRYKER
 - TestRuns

The DVMSimulationResults directory structure also contains command files used to run the simulations/models and reports generated from the runs.

DVMSimulationResults

- CommandFiles
- Word
- XML

FILE NAMING CONVENTIONS

The user of the MPP is able to select a file name for the output report from either the CM or the DVM. This file name is also used to track the results files.

There can be multiple maneuvers/tests selected from the DVM or CM. Multiple maneuvers of the same type can be selected. To keep track of the individual maneuver results, the data file names are appended with a three-letter acronym, and the run number. If the file name is “TestRuns” and a Straight Line Acceleration (SLA) maneuver is the third run, then the file TestRunsSLA_3.mat will appear in the sub-sub-heading TestRuns.

Some maneuvers allow the user to select multiple speeds and multiple values for other parameters as well. As an example, the Trapezoidal Bump maneuver allows the user to enter multiple speed and multiple bump height values. If either more than one speed or more than one bump height value is selected, the naming convention described above is appended with two numbers with the following format:

##

Where the first # is the bump height value and the second # is the speed value. The speed value is always the second # regardless of maneuver. For other maneuvers, the first # is the other potential multiple value parameter.

As an example, if the Trapezoidal Bump (TRP) maneuver is the fourth maneuver run, and there are three speed values and two bump heights, then the

following output files will exist (assuming TestRuns is still the file name selected):

TestRunsTRP_4_1_1.mat
 TestRunsTRP_4_1_2.mat
 TestRunsTRP_4_1_3.mat
 TestRunsTRP_4_2_1.mat
 TestRunsTRP_4_2_2.mat
 TestRunsTRP_4_2_3.mat

OUTPUT DATA STRUCTURES

A data structure format that is consistent between all simulation programs has been designed. It is highly unlikely that all the different simulation programs will be able to produce the required data structure so a method of converting the individual simulation program output formats to the required data structure format is required. This is not done in the DVM or CM, but instead separate translators are required to produce the desired format. The individual test runs are stored in a matlab file format (*.mat). The data is stored using SI units and the SAE vehicle axis (or coordinate) system (x- positive forward, z- positive down, y- positive right) will be used. A list of channel names is given in Table 1.

The test data that a user wants to examine with the DVM has to have the same format and file naming convention, units, and axis system as the data for the simulation output data structures.

If trailers are part of the vehicle model, then the trailer channel names will begin with TR#, with # - 1 for the first trailer, 2 for the second trailer. Currently just a single trailer is implemented (TR1)

Up to ten axles are allowed with up to three tires per axle side. Axles are numbered 1 through the total number of axles starting from the front going to the rear. Left and right side tires are designated with an L or R. It is assumed that multiple tires can be present on each axle. The tires are numbered from outside to inside starting with 1 for the outermost tire. Channel descriptors for tires will have the following format:

Descriptor_A#L(or R)# - ie: Axle number (front to rear), Axle side (L or R), tire number (outer to inner).

Row and seat locations use a similar number system. Row locations are numbered R1 through the total number of rows (front to rear or left to right) and seat locations are numbered S1 through the total number of seats (left to right or front to rear).

Table 1 - Channel Mnemonics, Units, and Definition

Mnemonic	Units	Definition
Time	(sec)	Simulation time
SteerWheelAngle	(rad)	Steering wheel angle
SteeringTorque	(Nm)	Steering torque
BrakeTravel	(m)	Brake pedal travel
BrakePedalForce	(N)	Brake pedal force

BrakePressure_A#LorR	(bar)	Brake pressure on Axle #, left or right side
ThrottlePos	-	Normalized throttle pos (0-1)
EngRPM	(RPM)	Engine speed
EngTorque	(Nm)	Engine torque
PowerTrainGear	-	Gear selected
PositionX	(m)	Sprung mass CG x position (global frame)
PositionY	(m)	Sprung mass CG y position (global frame)
PositionZ	(m)	Sprung mass CG z position (global frame)
VelocityX	(m/s)	Sprung mass CG x velocity (local frame)
VelocityY	(m/s)	Sprung mass CG y velocity (local frame)
VelocityZ	(m/s)	Sprung mass CG z velocity (local frame)
AccelerationX	(m/s/s)	Sprung mass CG x acceleration (local frame)
AccelerationY	(m/s/s)	Sprung mass CG y acceleration (local frame)
AccelerationZ	(m/s/s)	Sprung mass CG z acceleration (local frame)
RollAngle	(rad)	Sprung mass Roll Angle
PitchAngle	(rad)	Sprung mass pitch angle
YawAngle	(rad)	Sprung mass yaw angle
RollRate	(rad/s)	Sprung mass roll rate
PitchRate	(rad/s)	Sprung mass pitch rate
YawRate	(rad/s)	Sprung mass yaw rate
SideSlip	(rad)	Sprung mass side slip angle
SuspForceZ_A#LorR	(N)	Suspension force
SuspDeflectionZ_A#LorR	(m)	Suspension deflection
SuspAccelerationZ_A#LorR	(m/s/s)	Suspension acceleration
TireDeflectionZ_A#LorR#	(m)	Vertical tire deflection on Axle #, left or right side tire # (Tire frame)
TireForceX_A#LorR#	(N)	Fx tire force on Axle #, left or right side tire # (Tire frame)
TireForceY_A#LorR#	(N)	Fy tire force on Axle #, left or right side tire # (Tire frame)
TireForceZ_A#LorR#	(N)	Fz tire force on Axle #, left or right side tire # (Tire frame)
TireMomentZ_A#LorR#	(Nm)	Mz tire moment on Axle #, left or right side tire # (Tire frame)
PositionX_A#LorR	(m)	Global tire X position

#		
PositionY_A#LorR#	(m)	Global tire Y position
SlipAngle_A#LorR#	(rad)	Slip angle, Axle #, left or right side tire # (Tire frame)
CamberAngle_A#LorR#	(rad)	Camber angle, left or right side tire # (Tire frame)
SteerAngle_A#LorR#	(rad)	Steer angle, left or right side tire # (Tire frame)
WheelSpeed_A#LorR#	(rad/s)	Wheel speed, Axle #, left or right side tire # (Tire frame)
WheelSlip_A#LorR#	-	Wheel slip, Axle #, left or right side tire # (Tire frame)
TerrainHeight_A#LorR#	(m)	Terrain height, Axle #, left or right side tire # (Tire frame)
AccelerationZ_R#S#	(m/s/s)	Floor acceleration below row # and seat #

COMMAND FILES

Based on the values entered on the DVM (or CM) GUI for each maneuver, the DVM creates command files that are used to run the simulations. These command files are stored in the CommandFiles sub-directory under DVMSimulationResults:

DVMSimulationResults

- CommandFiles

These files are named using the Report Name, Maneuver Type (3 letter code), and the Maneuver Index Number (where the maneuver is on the Maneuver List). If the Report Name is "Default" and the third maneuver is a fishhook, then the command files will have the following name:

DefaultFSH_3.*

There are currently eight types of command files: steering, throttle, speed, brake, gear, K & C, terrain, and drawbar pull. The extensions for these different types of command files are:

```
Steering = .dsw
Throttle = .thr
Speed = .u
Brake = .brk
Gear = .gear
K & C = .kc
Terrain = *.ter
Drawbar Pull = *.dbp
```

The fishhook maneuver requires a steering and throttle profile, so for the Fishhook maneuver discussed above, two command files will exist:

DefaultFSH_3.dsw
DefaultFSH_3.thr

Some maneuvers allow the user to select multiple speeds and multiple values for other parameters as well. As an example, the Trapezoidal Bump maneuver allows the user to enter multiple speed and multiple bump height values. If the maneuver has the option for multiple speeds or other parameters, then the file naming convention will append an additional number onto the command file name: _#. For a speed command file the additional number will be the speed number and for other command files it will be the series number associated with the other multi-valued parameter.

As an example, the Trapezoidal Bump (TRP) maneuver requires a terrain profile and a speed command file. If the Trapezoidal Bump is the fourth maneuver run, and there are three speed values and two bump heights, then the following command files will exist (assuming Default is still the file name selected):

DefaultTRP_4_1.ter
DefaultTRP_4_2.ter
DefaultTRP_4_1.u
DefaultTRP_4_2.u
DefaultTRP_4_3.u

SIMULATION WRAPPERS

When the DVM (or CM run) button is selected, the MPP and DVM GUI data structures are used to make calls to simulation wrappers. There is a simulation wrapper for each simulation program that is supported. There is currently a VDANL and SimCreator wrapper. The wrappers are sent file names and locations for command files (speed, gear, throttle, braking, steering, K & C, etc.), vehicle model file locations, and output file names. A simulation wrapper is called for each combination of vehicle model and maneuver.

REPORTING OPTIONS

There are three reporting options: XML, Word, or Matlab® Figures. XML allows the report(s) to be viewed using a web-browser. If this option is selected, the web-browser is started and the report is displayed after all the runs/metric calculations have been completed. The Word option generates and saves a report in Microsoft® Word format. The associated files for either of the XML or Word options are named based on the user defined file name entered on either the DVM or CM GUI. The figures in the XML or Word reports are generated in Matlab. These figures can be generated as stand-alone windows using the Matlab Figures reporting option. The user can then use the individual figures as necessary (copy and paste into other programs, etc.)

Each figure generated has trace(s) for each vehicle model selected unless the simulation program for the vehicle model is not capable of completing the maneuver. Each trace is distinguished using different

colors. A legend is created using the simulation program and vehicle model names. The following acronyms are used:

SC = SimCreator
DA = DADS
VD = VDANL

If the HMMWV model for SimCreator is chosen, then the legend name is:

SC_HMMWV

If an Additional Parameter file named "Heavy" is selected, then the legend name is:

SC_HMWWV_Heavy

This type of labeling is also used for the tables.

The legend color for the given vehicle model/simulation program combination will remain the same for a given report.

The reports are stored under the DVM (or CM) SimulationResults directory. There are separate directories for Word and XML results.

DVMSimulationResults

- Word
- XML

DATA SHARED BETWEEN THE CM AND DVM

Some of the parameters calculated in the CM can be used by the DVM. This is done using an output file from the CM as input to the DVM. The output file is called CMparams4DVM.mat. This file is output after the CM is run. This file is located in the appropriate model directory for each model run by the CM. When the DVM is run, a check is made to see if the file exists for each model being run by the DVM. If it does exist, then it is opened and the parameters are used where appropriate.

The CM will create a separate CMparams4DVM.mat file for each Simulation and Model, and Additional Parameter combination selected on the MPP GUI. If the vehicle is a SimCreator model called "HMMWV" then there will be a file called "CMparams4DVM.mat" listed in the "HMMWV" directory after the CM is run. This file would be given in the following directory tree

Models

- SimCreator
 - HMMWV
 - parameter files for HMMWV
 - CMparams4DVM.mat

If a user selects an Additional Parameter file on the MPP GUI to use when running the CM, then a CMparams4DVM.mat file will be created for the

Additional Parameter file. This file will be saved in a subdirectory under the model directory that will have the same name as the Additional Parameter file selected for use in the MPP GUI. As an example, if the user selects the HMMWV_light.in as an additional parameter file for use with the SimCreator HMMWV model, then the CMparams4DVM.mat file would be saved in the following structure.

Models

- SimCreator
 - HMMWV
 - parameter files for HMMWV
 - HMMWV_light
 - CMparams4DVM.mat
 - AdditionalParameters
 - HMMWV_light.in
 - HMMWV_heavy.in

An example of how this file is used is the calculation of the understeer gradient for the Slowly Increasing Steer maneuver in the DVM. The understeer gradient calculation requires that the steering ratio and the wheelbase of the vehicle be known (for a front axle only steered vehicle with two axles and only one tire for each axle location). These values are determined using the CM Static Test and Steering Ratio test. These values are saved in the CMparams4DVM.mat file. This file can be opened by the DVM (if it exists). Just prior to the understeer gradient calculation, the DVM checks for the existence of the CMparams4DVM.mat file. If it exists, it is opened and it checks for the existence of the steering ratio and wheelbase values. If they exist, then they are used to calculate the understeer gradient, otherwise no understeer gradient is determined.

CONCLUSIONS

This paper presented the development a vehicle model/simulation tool called the Model Post Processor (MPP). The developed tool allows a vehicle dynamicist to select vehicle models from a variety of simulation programs and to evaluate/compare/contrast them using static vehicle metrics and/or using the results from dynamic vehicle maneuvers to evaluate their simulation and/or model. This tool allows the user to compare different vehicle models for the same simulation, vehicle models for different simulations, and to compare vehicle models to physical test data.

CONTACT

J. Gavin Howe, Senior Research Engineer, Systems Technology, Inc., gavin@systemstech.com, (310) 679-2281 (ext. 39).

DEFINITIONS, ACRONYMS, ABBREVIATIONS

MPP: Model Post Processor

DVM: Dynamic Vehicle Metrics

CM: Consistency Metrics

Vehicle Dynamics: The study of how and why forces are produced by and on a vehicle and the resulting vehicle motion (acceleration, braking, handling, and ride).

Simulation (Package): A computer algorithm with equations of motion that define the behavior of a vehicle and/or its components to control inputs.

Vehicle Model: A set of parameters that define a vehicle for a given simulation package.

REFERENCES

- 1 - Olley, M., "Road Manners of the Modern Car," The Institution of Automobile Engineers, 1946-47.
- 2 - Segal, L., "Theoretical Prediction and Experimental Substantiation of the Response of the Automobile to Steering Control," Proceedings of the Institution of Mechanical Engineers, 1956-57.
- 3 - Heydinger, G. J., Garrott, W. R., Chrstos, J. P., Guenther, D. A., "A Methodology for Validating Vehicle Dynamics Simulations," SAE Paper No. 900128, 1990 International Congress and Exhibition, February-March 1990.
- 4 - Bernard, J. E., Clover, C. L., "Validation of Computer Simulations of Vehicle Dynamics," SAE Paper No. 940231, 1994 SAE International Congress and Exhibition, February 1994.
- 5 - "High Fidelity Ground Platform and Terramechanics Modeling (HGTM) Army Technology Objective (ATO), Appendix A Test Data Analysis Plan," U.S. Army Tank Automotive Research, Development & Engineering Center, Cold Regions Research Laboratory, Geotechnical Structures Laboratory, November 2006.

APPENDIX

The screenshot displays the 'Model Post Processor' application window. The title bar includes the application name and standard window controls. The menu bar contains 'Form Management' and 'Help'. The main interface is divided into three sections: 'Model Directory' with a text field showing 'c:\MPP\Models' and a 'Browse' button; 'Simulation/Model Selection' which contains a table with columns for 'Simulation', 'Model', and 'Additional Parameters'; and 'MPP Control' with four buttons: 'Save MPP', 'Load MPP', 'Consistency Metrics', and 'Dynamic Vehicle Metrics'.

	Simulation	Model	Additional Parameters
<input type="checkbox"/> Run 1	-None		
<input type="checkbox"/> Run 2	-None		
<input type="checkbox"/> Run 3	-None		
<input type="checkbox"/> Run 4	-None		
<input type="checkbox"/> Run 5	-None		
<input type="checkbox"/> Run 6	-None		
<input type="checkbox"/> Run 7	-None		
<input type="checkbox"/> Run 8	-None		

Figure 2 - MPP GUI

Consistency Metrics [Minimize] [Maximize] [Close]

Form Management Help

Test Selection

Kinematic Heave

Max Tire Load (% Veh Wgt)

Ramp Rate (m/s)

Additional Description

Add Test

Test List

Remove Test

Run Status

File Options

☒ Matlab ☐ XML ☐ Word File Names

Output Units

Fit Data? ☐ Yes ☒ No

Legend? ☒ Yes ☐ No

CM Control

Save CM Load CM Run CM

Run Type? ☒ All ☐ Selected

Overwrite Tests? ☒ Yes ☐ No

Stop CM Close Matlab Plots

Figure 3 – CM GUI

Dynamic Vehicle Metrics

Form Management
Help
RMS Course Values

Maneuver Selection

Slowly Increasing Steer

Maneuver Inputs

Vehicle Speed (m/s)

Max Steer Angle (rad)

Steer Rate (rad/s)

Gear Number

Additional Description

Add Maneuver

Maneuver List

Remove Maneuver

Maneuver Inputs

Run Status

File Options

☒ Matlab
☐ XML
☐ Word

File Names

Default

Output Units

Metric

Use Plot Limits?
☐ Yes
☒ No

Legend?
☒ Yes
☐ No

Display Only Summary Plots?
☐ Yes
☒ No

DVM Control

Save DVM
Load DVM
Run DVM

Run Type?
☒ All
☐ Selected

Overwrite Maneuvers?
☒ Yes
☐ No

Stop DVM
Close Matlab Plots

Figure 4 – DVM GUI